

Richtlijnen Bachelor Eindwerk Informatica

J. Broeckhove

December 3, 2013

1 Inleiding

1.1 Inhoud en vorm

1.1.1 Eindcompetenties

De eindcompetenties van het Bachelor eindwerk situeren zich in drie competentieclusters:

Vakoverschrijdende, wetenschappelijke competenties

- Je kan een wetenschappelijke probleemstelling analyseren, diverse oplossing formuleren en tegen mekaar afwegen.
- Je kan je kennis van diverse Informatica disciplines (programmeren, datastructuren, netwerken, ...) inzetten om een oplossing te ontwerpen.
- Je kan de oplossing operationaliseren: welke bestaande componenten gebruiken, welke nieuw realiseren, welke kennis heb ik daarvoor nodig en hoe zal ze verworven worden, hoe zal ik precies het nieuwe systeem realiseren.

Werken in teamverband

- Je kan in een groep van 4 of 5 werken en in overleg werkzaamheden plannen en uitvoeren.
- Je kan met een kritische maar constructieve benadering zowel voor je eigen werk alsdat van anderen doorlichten.
- Je kan omgaan met project deadlines en het werkvolume inschatten en verdelen onder groepsleden.

Communicatie

- Je kan vlot de schriftelijke documentatie voor software opstellen.
- Je kan mondeling rapporteren over werkorganisatie en -vordering.
- Je kan een uiteenzetting geven aan derden over het gevoerde project.

Software ontwikkeling

- Je kan een gestructureerd softwareontwikkelingsproces doorlopen.
- Je kan tools (versie controle, programmabouw, testen, UML, e.d.) leren gebruiken en inzetten in het project.

1.1.2 Inhoud

Het eindwerk voor de studenten Bachelor Informatica is een vakoverschrijdend groepsproject waarbij een softwareoplossing aan een concreet wetenschappelijk probleem gegeven wordt. De probleemomschrijving wijzigt van jaar tot jaar en wordt ook pas in de loop van het jaar meegedeeld, maar de aanpak van het groepsproject verloopt telkens volgens dezelfde lijnen. Die aanpak omvat onderzoek van de probleemstelling, het voorstellen van een oplossing, de software realisatie bestaande uit ontwerpen, coderen, testen, documenteren, en ten slotte de rapportering inzake de voorgestelde oplossing.

Tijdens het theoretisch luik van de cursus krijgt de student een inleiding in de probleemstelling en een overzicht van de vereisten inzake:

- het functioneren in groep
- de periodieke en finale rapportering
- het softwareontwikkelingstraject
- de tools die gehanteerd moeten worden
- de mijlpalen die gerespecteerd moeten worden
- de werkproducten die opgeleverd moeten worden

Niet alleen de opgeleverde werkproducten (code, documenten, presentaties, ...) maar ook de aanpak van het project is van belang. Er zijn vooraf specifieke termijnen vastgelegd waarop project-producten zoals ontwerpdiagrammen of prototype code opgeleverd moeten worden evenals momenten waarop mondeling of schriftelijk gerapporteerd moet worden. Het respecteren van die termijnen is een onderdeel van de eindevaluatie (zie sectie 2.4 voor meer details).

Het gaat in het project steeds om *functionele kennis* dewelke de student moet helpen in het realiseren van het project. De uitvoering van het project dwingt de student om kennis in vaardigheden om te zetten e.g. kennis van syntax omzetten naar programmeren, kennis van een software proces omzetten naar het uitvoeren ervan, enzovoort. Dat vereist volgende stappen:

- het lezen van uiteenzettingen in de nota's of in andere bronnen
- het bestuderen van voorbeelden, ze nabootsen, omvormen, kortom experimenteren
- interageren over bevindingen met medestudenten
- toepassen van vaardigheden in het kader van de projectopdracht

Deze cyclus wordt telkens opnieuw doorlopen voor deelonderwerpen en zo breidt de student zijn competenties stelselmatig uit.

1.1.3 Structuur

In lijn met de focus van het eindwerk op het vakoverschrijdend oplossen van een probleemstelling met bijbehorende implementatie van de oplossing in software, wordt de portfolio van op te leveren documenten als volgt gestructureerd :

Projectvisie document

Een beschrijving van het project, maximum twee pagina's lang. Het is het resultaat van de voorstudie en van de eerste inzichten over de manier waarop de voorgestelde oplossing zal tegemoetkomen aan de probleemstelling.

Use case document

De documenten uit de inception fase moeten ingediend worden, in één geïntegreerd document. Hierbij dient voor één use case een “fully-dressed” beschrijving met een Cockburn template en een UML-use case diagram opgesteld te worden. Voor overige use cases volstaat een use-case diagram ofwel een korte beschrijving.

Domein analyse document

Het domein model wordt uitgewerkt aan de hand van de Collaboration Responsibility Cards (CRC) techniek. De belangrijkste entiteiten (UML klasse diagram) en interacties (UML sequence of collaboration diagram) uit het probleemdomein worden weergegeven.

Architectuur document

Het architectuur document geeft de architectuur van de oplossing weer vanuit de *logische view*, via een UML package diagram, en vanuit de *deployment view*, via een UML deployment diagram en een component diagram.

Test plan

Er wordt een weloverwogen, geautomatiseerde testmethode gehanteerd tijdens de implementatie activiteit. Deze tests omvatten unit tests en scenario tests. Ze worden uitgevoerd door de continuous integration server. Het test plan beschrijft de testscenarios, bevat een oplijsting van de functionaliteiten die worden getest (en eventueel welke niet) en geeft aan hoe de tests geïmplementeerd zijn.

High-level Design

Publicatie van het design op hoog niveau. Hieronder verstaan we een klasse diagram met de klassen die een sleutelrol spelen bij het realiseren van de use cases, en hun onderlinge band eventueel uitgedrukt via een object interaction diagram.

Project verslag

Het verslag dient vooral de onderliggende visie en concepten gehanteerd bij de uitwerking van het project uiteen te zetten en te verantwoorden. Belangrijk zijn daarbij de afwegingen die tussen alternatieven werden gemaakt. De ordegrrootte van het verslag (exclusief figuren, document werkproducten in bijlage e.d.) is minimum tien bladzijden.

1.1.4 Vorm

Alle documenten dienen opgeleverd te worden in PDF formaat. Het staat de projectgroep vrij om een taalkeuze te maken (Engels of Nederlands) voor deze documenten. Voor de oplevering van het projectverslag dienen de richtlijnen uiteengezet in "A Guide to Technical Report Writing, IET, 2012" gevolgd worden. Deze gids is opgenomen als appendix in de cursusnota's.

Bij materiaal (algoritme, design, code, ...) geput uit externe bron (boek, internet, andere groep, ...), al dan niet na verdere bewerking, geheel of gedeeltelijk, opgenomen in een werkproduct (document, code, website, ...) moet een duidelijk zichtbare *bronvermelding* aangebracht worden. Het nalaten van dergelijk vermelding geldt als fraude en zal bestraft worden in overeenstemming met het examenreglement. De groepsleden hebben hier een collegiale en collectieve verantwoordelijkheid om na te gaan dat de stelregel voor het geheel van het opgeleverde werk gerespecteerd wordt.

2 Procedure

2.1 Bepaling van het onderwerp

De probleemstelling die in de Bachelor proef wordt aangebracht wijzigt jaarlijks en wordt tijdens de contactmomenten uiteengezet. De oplossing zal een aantal componenten omvatten. Elk van deze componenten wordt door de lesgever geassocieerd met een set van *minimale requirements* en *maximale requirements*. De projectgroep zal met de lesgever een negotiatie doorlopen om vast te leggen welke componenten doorheen het academiejaar zullen worden uitgewerkt en welke set requirements per component wordt ingepland voor implementatie. Op deze manier wordt de vaardigheid aangebracht om implementatie en analyse opdrachten af te schatten naar werklust.

2.2 Begeleiding

Naast begeleiding die op individuele aanvraag aan de student of projectgroep wordt verzorgd in de loop van het academiejaar, worden volgende types contactmomenten ingepland:

- Overzicht projectverloop en -onderwerp : een sessie waarin de lesgever de doelstellingen, verwachtingen, evaluatie en werkprocessen toelicht van het eindwerk.
- Workshops waarin technische onderwerpen worden behandeld, noodzakelijk voor de succesvolle uitwerking van het project (e.g. multi-threading, inter-process communicatie, socket programmatie).

- Functioneringsgesprekken : Er worden vier gespreksmomenten georganiseerd waarin het huidige verloop van het eindwerk met de groep wordt besproken en feedback wordt gegeven t.a.v. opgeleverde werkproducten. De sjablonen voor deze gesprekken zijn terug te vinden in de cursusnota's.

2.3 Tijdsschema

In de werkkalender, hieronder opgegeven per semester en ingedeeld per lesweek, zijn de contactmomenten, de activiteiten van de groep en de mijlpalen opgenomen. De kalender correspondeert met de planning van academiejaar 2013-2014 en is als voorbeeld in deze richtlijn opgenomen.

Voor wat de activiteiten betreft is dit een aanduiding van de belangrijkste, maar niet noodzakelijk de enige, activiteit voor die week. Zo wordt in lesweek drie van het eerste semester "Web paginas opmaken" vermeld als activiteit. Bedoeld wordt het programmeren van de website die de groep opstelt om zichzelf voor te stellen en om zijn werkproducten aan de begeleiders kenbaar te maken. Het is duidelijk dat ook in latere weken enige webprogrammatie zal plaatsvinden om de website te hertekenen of uit te breiden.

De opgegeven mijlpalen (*milestones* in het Engels) van het project verwijzen naar het opleveren van werkproducten (*artifacts* in het Engels) zoals documenten, UML diagrammen, source code, webpaginas, test resultaten en zo voort. De oplevering van artifacts van een mijlpaal gebeurt ten laatste op zondag, 23h55, van de aangeduide week. Artifacts worden opgeleverd door ze onder te brengen in de *repository* voor versiebeheer van de groep.

Een project kan meerdere soorten deadlines hanteren. In de planning van het project onderscheiden we twee types:

Externe deadlines worden gekenmerkt door

- vastgelegd in samenspraak met de opdrachtgever
- geassocieerd met de oplevering van producten
- vastgelegd als mijlpalen in de planning
- harde deadlines die niet verschoven kunnen worden

Interne deadlines worden gekenmerkt door

- vastgelegd na overleg binnen de projectgroep
- worden enkel in consensus gewijzigd
- missen heeft een belangrijke signaalfunctie: één, meerdere of alle groepsleden presteren onvoldoende, de interne planning is onrealistisch, de werkverdeling is onevenwichtig, ...; remediëring is belangrijk

Problemen in verband met interne deadlines moeten zeker aan de orde gesteld worden tijdens de functioneringsgesprekken. Het is belangrijk deze deadlines expliciet neer te schrijven, bijvoorbeeld op de website, om mis-communicatie te vermijden en verantwoordelijkheden duidelijk vast te leggen. Het niet naleven van de planning vertaalt zich in een minder goede beoordeling bij de eindevaluatie.

Lesweek	Lessen	Activiteit	Mijlpalen
1			
2	- Overzicht Projectverloop en -onderwerp	Eerste contact	- Groep gemeld
3	- Workshop Multi-threading	- Groep samenstellen - Web paginas opmaken	- VM in gebruik - Website operationeel
4	- Workshop IPC	- Achtergrondstudie project onderwerp - Studie van tools	- Repository & CI server - MT prototype
5	- Workshop sockets	- Achtergrondstudie project onderwerp - Studie van tools	- IPC prototype
6		- Use cases	- Socket prototype
7		- Use cases - Domein analyse	- Projectvisie document
8	- Functionering & feedback	- Domein analyse	- Use Case document
9		- Domein analyse - Architectuur analyse	
10		- Architectuur - Implementatie prototypes	- Domein analyse document
11	- Functionering & feedback	- Architectuur - Implementatie prototypes	
12		- Implementatie prototypes	- Architectuur document
13		- Implementatie prototypes	
Reces			

Figure 1: Kalender voor het eerste semester.

Lesweek	Lessen	Activiteit	Mijlpalen
1		- High-Level Design - Testframework	- Architectuur prototype
2	- Functionering en feedback	- High-Level Design - Implementatie - Testframework	
3		- High-Level Design - Implementatie - Testing	- Test plan - Tests operationeel
4		- Implementatie - Testing	- High-Level Design
5		- Implementatie - Testing	
6		- Implementatie - Testing	
7		- Implementatie - Testing - Demo alpha-versie	- Alpha-versie
8	- Functionering en feedback	- Implementatie - Testing	
9		- Implementatie - Testing	
10		- Implementatie - Testing	- Beta versie
11	- Feedback	- Testing	
12		- Finale tests - Finale inspectie alle artifacts	
Examen periode		- Oplevering 1 week voor examen - Presentatie en demo	- Alle project artifacts - Presentatie document

Figure 2: Kalender voor het tweede semester.

De mijlpalen in het schema worden als volgt gedefinieerd:

Groep gemeld

De groepen zijn samengesteld – uit vier of vijf leden – en hebben een naam gekozen en een contactpersoon aangeduid. Deze informatie wordt via email aan de lesgever en begeleider gecommuniceerd. Groepsleden wijzen één van de leden aan als contactpersoon ten aanzien van de projectbegeleiders. De contactpersoon heeft geen leidinggevende functie in de groep, maar staat in voor de communicatie en neemt contact op met de begeleider in geval van moeilijkheden in de groep, problemen met deadlines, ...

VM in gebruik De virtuele machine waar de groep mee zal werken is in gebruik genomen, accounts aangemaakt, afspraken gemaakt voor verantwoordelijkheden bij systeembeheer, afspraken voor backup, en zo voort.

Website operationeel

De website van de groep is operationeel en bevat een voorstelling van elk groepslid. De site mag meer informatie bevatten dan via deze opdrachtoomschrijving geëist wordt. In het raam van de zelfstudie kan men interessante bronnen aangeven, of syntheses die men gemaakt heeft toevoegen, ...

Repository operationeel

De repository voor version management over alle project artefacts (code, documentatie, diagrammen, test drivers, ...) moet operationeel zijn. Verwijzing naar de repository moet in de website geïntegreerd zijn.

CI server operationeel

De continuous integration (CI) server die gebruikt zal worden om continu en herhaaldelijk testen te voeren moet operationeel zijn. De overzichtspagina's van het project gegenereerd door de CI server moeten zichtbaar zijn en in de website van de groep ingewerkt.

Projectvisie document

Een beschrijving van het project, maximum twee pagina's lang. Het is het resultaat van de voorstudie en van de eerste inzichten over de manier waarop de voorgestelde oplossing zal tegemoetkomen aan de probleemstelling.

GUI test ingevoerd Met het Sikuli framework worden GUI testen ingevoerd en geïntegreerd in de bestaande test infrastructuur.

Use case document

De documenten uit de inception fase moeten ingediend worden, in één enkel pdf-bestand. Hierbij dient voor één use case een “fully-dressed” beschrijving met het Cockburn template (zie cursus nota's) en een UML-use case diagram opgesteld te worden. Voor overige use cases volstaat een use-case diagram ofwel een korte beschrijving.

Domein analyse document

Het domein model wordt uitgewerkt aan de hand van CRC techniek [1]. De belangrijkste entiteiten (UML klasse diagram) en interacties (UML sequence of collaboration diagram) uit het probleemdomein worden weergegeven en opgeleverd in pdf formaat.

Architectuur document

Het architectuur document moet, telkens in een pdf-bestand, gepubliceerd worden. Het gaat hier om een weergave van de architectuur vanuit de logische view, via een UML package diagram, en om een weergave vanuit de deployment view, via een UML deployment diagram en een component diagram.

Architectuur prototype Een prototype code van de applicatie(s) met daarin alle essentiële architectuur elementen moet gerealiseerd zijn. De prototypes moeten gedemonstreerd kunnen

worden. De functionele inhoud van het prototype speelt geen enkele rol: het moet enkel de belangrijke stappen op architecturaal gebied (communicatie, synchronisatie, signalen, ...) implementeren.

Test plan

Er wordt een weloverwogen, geautomatiseerde testmethode gehanteerd tijdens de implementatie activiteit. Deze tests omvatten unit tests en scenario tests. Ze worden uitgevoerd door de continuous integration server. Het test plan beschrijft de testscenarios, bevat een oplist- ing van de functionaliteiten die worden getest (en eventueel welke niet) en geeft aan hoe de tests geïmplementeerd zijn.

Tests operationeel Een overzichtelijk rapport wordt gegenereerd m.b.t. welke testen slagen en welke falen, of welke testen op dit moment nog niet toegepast worden. Er wordt niet vereist dat op ieder moment van de implementatie alle testen slagen. Sommige testen zijn misschien nog niet van toepassing, voor andere is de functionaliteit nog maar partieel geïmplementeerd.

High-level Design

Publicatie van het design op hoog niveau in pdf-formaat. Het ontwerp bestaat uit een klasse diagram met de klassen die een sleutelrol spelen bij het realiseren van de use cases, en hun onderlinge verbanden eventueel uitgedrukt via een object interaction diagram.

Alpha versie

Een alpha versie moet beschikbaar zijn en gedemonstreerd kunnen worden. Een eerste versie van de gebruikershandleiding en projectverslag moeten beschikbaar zijn.

Beta versie

Een beta versie moet beschikbaar zijn en gedemonstreerd kunnen worden. De gebruiker- shandleiding en projectverslag moeten grotendeels afgewerkt zijn.

Alle project artifacts

Finale oplevering van het project. Alle testen moeten succesvol doorlopen worden. De website, de code met inline documentatie, en de gebruikershandleiding moet volledig in orde zijn.

Presentatie

Presentatie van het project, met daarbij een demonstratie van de functionaliteit aan de hand van het doorlopen van een aantal use case scenarios. Je mag daarbij slides, opgenomen in een pdf-bestand, gebruiken en alle werkproducten van het project, waaronder de applicatie zelf.

2.4 Beoordeling

2.4.1 Inleiding

Als algemeen uitgangspunt geldt voor dit opleidingsonderdeel:

De evaluatie toetst welke vaardigheden je daadwerkelijk in het project toegepast hebt.

De quotering komt tot stand als product van de individuele score via peer assessment (beoordeling door collega's in de groep) en de groepsscore toegekend door een panel:

individuele score = pa factor x groepsscore

met een maximum van 20 op 20 punten. De wijze waarop de beide onderdelen, peer assessment factor en groepsscore tot stand komen wordt hieronder uiteengezet.

2.4.2 Groepsscore

De groepsbeoordeling wordt opgesteld door een jury met lesgever, begeleider en eventueel externe beoordelaars. Ze slaat op het geheel van de geleverde werkproducten door de groep en wordt opgesteld door een jury met lesgever, begeleider en externe beoordelaars. Daarbij worden de volgende criteria toegepast inzake groepswerking en opgeleverde werkproducten (deze criteria moeten natuurlijk cumulatief geïnterpreteerd worden):

tussen 10 en 13 op 20

- analyse en ontwerp: er is een use case document plus heldere ontwerpdocumentatie, zodat een collega-student dit project zou kunnen overnemen
- implementatie en tests: er is een werkend systeem waarvan aangetoond kan worden (d.m.v. tests) dat het op functioneel vlak voldoet aan de minimum behoeften omschreven in het lastenboek
- de basisregels ivm projectvoering (groepssamenwerking, gebruikte tools, werkvolume) werden gerespecteerd

tussen 13 en 15 op 20

- strategische keuzes tijdens het eindwerk worden verantwoord in het projectverslag
- de gestelde doelen werden op planmatige wijze bereikt met respect van de termijnen
- de planning werd waar nodig tijdig en doelbewust bijgestuurd

tussen 15 en 17 op 20

- de uitgebreide behoeften opgenomen in het lastenboek werden gerealiseerd
- de kwaliteit van de opgeleverde werkproducten is uniform zeer goed
- de wijze van projectvoering en het gepast gebruik van tools getuigt van gedegen vakken-nis

meer dan 17 op 20

- het project werd uitstekend verdedigd
- de kwaliteit van de opgeleverde werkproducten is uniform uitstekend en getuigt van inzicht in de gestelde problematiek en de uitgevoerde oplossingen
- er werd een uitstekend projectverslag geschreven

2.4.3 Peer assessment

De individuele beoordelingen worden geleverd door een peer assessment, dit wil zeggen beoordeling door collega's in de groep. Deze verloopt volgens onderstaand model:

1. Je beoordeelt enkel de andere groepsleden, niet jezelf.
2. Oordeel in eer en geweten en hou rekening met volgende criteria:
 - Participatie in discussies en aanbrengen van zinvolle ideeën
 - Constructieve opstelling bij werkverdeling, respect voor deadlines
 - Bruikbaarheid en kwaliteit van de opgeleverde artefacts
 - Hulpvaardigheid, constructieve kritiek, zin voor initiatief
 - Omvang en belang van de bijdrage tot het groepswerk
3. Je beschikt over $10 * (aantalgroepsleden - 1)$ punten, te verdelen over de overige groepsleden. Gebruik al je punten op. Wanneer je een groepslid meer dan 10 punten geeft, schrijf dan het criterium op waarop dat groepslid het best gepresteerd heeft. Geef je een groepslid minder dan 10 punten, schrijf dan het criterium op waarop dat groepslid het meest ondermaats gepresteerd heeft.
4. Ieder groepslid behaalt zo een *totaalscore* door de scores die hij kreeg van de andere groepsleden te sommeren. Genormaliseerd leidt dat tot de assessmentfactor $totaalscore / 10 * (aantalgroepsleden - 1)$. Deze factor geeft weer hoe de groepsleden de relatieve bijdragen tot het groepswerk inschatten.

5. De beoordeling is geheim en wordt niet aan de overige groepsleden meegedeeld. Enkel bij extreme variaties in de beoordeling met daarbij gebrekkige motivaties kan deze beoordeling gewijzigd worden door de projectleiders, na overleg met de betrokkenen.

Een voorbeeld zal het schema verduidelijken. Neem een groep van vijf personen die een groepsbeoordeling van 14/20 heeft gekregen. De leden vullen hun assessment in en dat geeft onderstaande tabel.

groepslid	A	B	C	E	F
A		12	8	11	9
B	10		9	11	10
C	11	11		9	9
E	10	10	10		10
F	9	12	9	10	
totaal	40	45	36	41	38
pa factor	1.0	1.125	0.9	1.025	0.95
quotering	14	16	13	14	13

Horizontaal leest men de punten uitgebracht door ieder van de leden. Verticaal leest men dus de punten ontvangen door ieder van de leden. Dat sommeert tot de totaalscores (op de zesde rij). Relatief (deling door 40 in dit geval) geeft dit de assessmentfactor in de zevende rij. Wanneer deze vermenigvuldigd wordt met de groepsbeoordeling (14 op 20 in dit geval) dan krijgt men de quotering voor elk groepslid. In dit geval werd die reeds omgezet naar gehele punten.

2.4.4 Kwaliteitscriteria

Documenten

De documentatie dient telkens zo volledig mogelijk te zijn. Het is heel moeilijk om harde criteria op te leggen. Zo kan bijvoorbeeld een fully-dressed use case 1 tot 10 pagina's groot zijn, afhankelijk van de complexiteit van de use case. De kernvraag die telkens moet worden gesteld is of aan alle aspecten, alternatieven, randgevallen en mogelijke uitzonderingen gedacht is. Diagrammen worden, naast hun volledigheid, ook op hun semantische correctheid geëvalueerd. Soberheid en overzichtelijkheid zijn zowel voor de teksten als de diagrammen belangrijke eigenschappen.

Code

De algemene stelregels voor goede software die in de opleidingsonderdelen programmeren en aanverwante aangeleerd worden moeten hier ook opgevolgd worden. Daarnaast bestaan er tools die de kwaliteit van code automatisch beoordelen en een eventueel probleem situeren. Het gebruik van dergelijke tools is verplicht.

Tests

Tests moeten zo volledig mogelijk zijn. Dit is dan ook het voornaamste beoordelingscriterium. Dit wil niet zeggen dat elk mogelijk scenario moet worden getest. Indien een goede verantwoording kan worden geven waarom bepaalde functionaliteiten of deelaspecten ervan niet getest worden of slechts met een beperkte dekking getest worden, is dit ook in orde.

Website

Overzichtelijkheid in de navigatie (ongeacht de gebruikte browser) is het primaire kwaliteit-aspect. Daarnaast zijn soberheid en uiteraard volledigheid inzake de geleverde informatie een pluspunt.

Gebruikershandleiding

De gebruikershandleiding mag beperkt zijn (ordegrootte tien bladzijden alles inclusief) en mag in de code documentatie (gegenereerd via doxygen) geïntegreerd worden.

Projectverslag

Het verslag dient vooral de onderliggende visie en concepten gehanteerd bij de uitwerking van het project uiteen te zetten en te verantwoorden. Belangrijk zijn daarbij de afwegingen die tussen alternatieven werden gemaakt. De ordegraote van het verslag (exclusief figuren, document werkproducten in bijlage e.d.) is minimum tien bladzijden.

Presentatie

De presentatie dient overzichtelijk gestructureerd te zijn, met verzorgde slides. De uiteenzetting dient met een verzorgd taalgebruik te gebeuren. Inhoudelijk moet je zo snel mogelijk terzake komen.

References

- [1] David Bellin and Susan Suchman Simone. *The CRC Card Book*. Addison Wesley Longman, Inc., Reading, Massachusetts, 1997.